

Agile Webentwicklung mit Ruby on Rails

Sommersemester 2012

Blatt 6

Im Ordner `blog` finden Sie die in der Vorlesung erzeugte Applikation *Blog*. Alle Aufgaben sollten nacheinander in derselben Applikation bearbeitet werden.

Aufgabe 1: User-Website

Fügen Sie zur Ressource `User` ein Attribut `url` hinzu, welches die private Website des entsprechenden Nutzers repräsentiert. Führen Sie alle dafür notwendigen Anpassungen in Model, View, Controller und Presenter durch.

In der Show-View des Nutzers soll die Website als Link anklickbar sein. Falls für den Nutzer keine Website-URL bekannt ist, so wird *n/a* angezeigt.

Aufgabe 2: Article Presenter und Tag-List

Die Tag-Liste eines `Article` soll nur angezeigt werden, wenn tatsächlich auch Tags vorhanden sind. Andernfalls sollte in der View (abgesehen vom Formular natürlich) nichts auftauchen, das auf die Existenz von Tag schließen lässt. Weiterhin sollen die Tags als Links anklickbar sein und zur Show-View des jeweiligen Tags führen. Vergessen Sie nicht das, `Partial _article.html.erb` ebenfalls geeignet zu erweitern.

Implementieren Sie eine geeignete Presenter-Klasse für die Ressource `Article` und integrieren Sie diese dann in die Applikation.

Der Helper `content_tag`¹ könnte möglicherweise hilfreich sein.

¹<http://api.rubyonrails.org/classes/ActionView/Helpers/TagHelper.html>

Aufgabe 3: Presenter-Refactoring

Schreiben Sie eine Klasse **BasePresenter**, welche als Oberklasse für die anderen Presenter-Klassen dient. Extrahieren Sie gemeinsame Funktionalität aus den vorhandenen Presentern in diese Oberklasse.

Fügen Sie weiterhin eine Presenter-Klasse für die Ressource **Category** hinzu.

Implementieren Sie nun für alle Ressourcen an geeigneter Stelle Methoden, um die Aktions-Buttons in den Show- und Index-Views zu erzeugen. Passen Sie außerdem die Views entsprechend an.